Abstract—Recently we have presented transistor channel models of biological channels and the resulting implementation towards building spiking nodes, synapses, and dendrites. We have also discussed how to build reconfigurable dendrites using programmable analog techniques. With all of this technology components available, we begin to address the question of the computation model possible using a dendrite element, as well as a network of dendrite elements. We will discuss the connection between a dendrite element and a Hidden Markov Model (HMM) classifier branch, as well as a network of dendrites and somas to create an HMM classifier typical of what is used in speech recognition systems. We present simulation and experimental results for the branch elements; we also present initial results for a small dendrite based classifier structure to show the similarities to the HMM paradigm.

I. BIOLOGICALLY INSPIRED SILICON CLASSIFICATION

Hidden Markov Model (HMM) classifiers are key computations for speech recognition, as well as classification of other sensory modalities [1]. Analog implementation of these approaches is critical for an integrated classifier system. For example, an analog speech recognition system for phonemes [2] would require an analog cepstrum computation for feature extraction [3], analog Gaussian Mixture Model (GMM) or Vector Quantization (VQ) computation for symbol identification [4], [5], and one or more HMM classifier blocks; the first two cases have identified programmable implementations. This paper develops an analog version for an HMM implementation.

We describe an HMM classifier model and its resulting programmable IC implementation, first for a single branch computing a single metric, and then for a set of computational elements with supporting circuitry for an HMM classifier. Our approach implements HMM state progressions in a direct approach, taking algorithmic inspiration from [6], and building tight networks based on floating-gate circuits. We will show the connection between HMM state progression and dendritic computation at a fundamental level, and extend these approaches to look at interactions between neurons (e.g. pyramidal cells) with dendritic computation. We will further make comparisons of the computational efficiency of these elements versus digital computation; computational efficiency is as the number of computations, measured as Multiply ACCumulates per second (MAC), for a given power budget.

II. HMM BRANCH COMPUTATION

A Hidden Markov Model (HMM) can be viewed as a state machine in which the states themselves are not observable, but an output, whose statistics are determined by the current state, is observable. For example, in using an HMM to model speech production the states are the desired utterance (phonemes and words) and the observations are features of the audio signal produced by the talker. The audio features are determined by the spoken word but they are randomly distributed since each time that same word is spoken it will sound a little different. For recognition problems, the goal is to estimate the underlying states of the state machine based on the observed outputs. For speech recognition, the HMM decoder takes as inputs the signal statistics or features and generates a probability of occurrence on any one of a set of speech symbols. These symbols can be grouped over multiple short windows to generate larger symbols, such as phonemes or words. The ongoing input train of symbols is used to map a path through a probability trellis for the larger blocks [1].

We want to revisit the mathematical modeling for HMM classification and translate the formulation to be more easily implemented in continuous-time analog hardware. Figure 1 shows the circuit design for the continuous-time analog HMM branch element, based upon the branch network typically used in speech recognition. The floating-gate transistors are programmed such that the diffusor network exhibits wave propagation. Diffusor networks, as originally presented by Boahen, et. al [7], were built to emulate second-order diffusive spreading networks. Programming the transistor elements directly allows a richer set of Spatial-Temporal dynamics. Floating-gate transistors set the conductance of each element individually, thereby cancelling mismatch and allowing a desired conductance to be programmed.
A. Mathematical Modeling

We reformulate a given discrete-time HMM branch into a continuous-time HMM block that can be modeled using programmable transistors. Starting from the discrete-time formulation corresponding to Fig. 1

\[ \phi_i(n) = b_i(n)((1 - a_i)\phi_i(n - 1) + a_{i-1}\phi_{i-1}(n - 1)), \]

we extend this formulation to continuous-time signals as [8]

\[ \tau \frac{\partial \phi_i(t)}{\partial t} + \left( \frac{1}{b_i(t)} - 1 \right) \phi_i(t) = a_i(\phi_{i-1} - \phi_i(t)), \quad (1) \]

where \( \phi_i \) represents the stored state at the \( i^{th} \) node, and the inputs \( b_i(t) \) represent the input probabilities or confidence of a symbol occurring at a given moment in time. For several nodes, the equation can be expanded in position (x) to first order, resulting in a wave propagating PDE, where \( a_i \) modifies the velocity of the resulting wave, and the input probabilities, \( b_i(t) \), set the decay rate of the \( \phi_i \) values.

B. Configurable IC implementation

For the implementation, we leverage progress in Large-Scale Field Programmable Analog Arrays (FPAA) to build prototype the IC needed to measure these devices. We use the Reconfigurable Analog Signal Processor (RASP), that we described elsewhere [9]; this IC that utilizes 56 CABS (an 8 x 7 array) and multi-level interconnect routing with various levels of analog computational granularity utilizing over 50,000 programmable analog elements. In all CABS, we have 3 capacitors, with 2 capacitors with one terminal to ground, 1 nFET transistor, 1 pFET transistor, 3 OTAs with programmable bias currents, 2 C^4 bandpass amplifiers with programmable bias currents, 1 maximum amplitude detector, and 1 minimum amplitude detector. We use a multi-level routing network to configure blocks, which includes global horizontal bus, a global vertical bus, and a local vertical bus. Figure 2 shows the RASP block diagram and resulting die photo of the working IC. Since the switch elements are floating-gate transistors, they can also be used as resistors, diffusers, current sources, and other computational elements.

III. BUILDING AN HMM CLASSIFIER NETWORK

The output of an HMM branch is a metric of the confidence of a sequence of events occurring in a rough sequence. We build an HMM network by comparing the resulting HMM branch outputs (Fig. 3); only branches that correspond to useful sequences would indicate the presence of a useful symbol. This comparison occurs using a Winner-Take-All (WTA) network, similar to the classic structure by Lazzaro [10] plus input cascode transistors to isolate the WTA behavior from the HMM branch. When a valid symbol is identified at the WTA output, the output triggers a starting signal for the inputs of all the HMM branches. We model this starting symbol as a current source that turns on when a digital pulse is applied; this approach is the simplest approach, yet fully extended to more complicated starting sequences. These structures can be directly compiled in the RASP IC processor and experimentally measured. We will present our simulation and experimental results for the branch elements; we also present our initial results for a small HMM classifier structure.

IV. EFFICIENCY AND SCALING OF HMM NETWORKS

Having demonstrated the HMM network computation, we now consider the computational efficiency of these networks in comparison to digital computational approaches. To digitally compute (1), we require roughly 3 MAC at a sample rate of 10kHz, dictated by sample rate typically used to capture dendritic behavior which would be similar for an acoustic classifier, thereby requiring 30kMACs per node. Timing needs to be preserved, which tends to indicate higher sampling rates. For the analog HMM node, the power is set by the largest input current, which we assume sets a corner frequency (f_{corner}) of 1kHz, setting the bias current (I_{bias}) and power dissipated (P) by

\[ I_{bias} = 2\pi f_{corner} C U_T, P = 2\pi C U_T f_{corner} V_{dd}. \]

where C is the node load capacitance, V_{dd} is the power supply, and we operate with subthreshold current levels (U_T = thermal voltage or kT/q). For our FPAA setup, we use V_{dd} of 2.4V, although the HMM structure can operate with a supply of few 100mV.
Fig. 3. Diagram of HMM computational structure for a yes / no decision structure. The core structure is a set of HMM state transitions computing a desired metric, implemented using programmable transistors in a diffusor structure. The starting signal for the network is modeled as a switchable current source, giving a response similar to a silicon synapse. The largest output metric at the end of each branch is selected using a standard WTA circuit.

<table>
<thead>
<tr>
<th>C</th>
<th>P</th>
<th>MMAC / $\mu$W</th>
<th>$I_{bias}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1pF (RASP IC)</td>
<td>390pW</td>
<td>460</td>
<td>160pA</td>
</tr>
<tr>
<td>30pF (config 0.35$\mu$m)</td>
<td>12pW</td>
<td>15000</td>
<td>5pA</td>
</tr>
<tr>
<td>10F (config 0.13$\mu$m)</td>
<td>7.8pW</td>
<td>23100</td>
<td>1.6pA</td>
</tr>
<tr>
<td>Digital</td>
<td>0pW</td>
<td>0.01 - 0.001</td>
<td></td>
</tr>
</tbody>
</table>

Table I

RESULITNG POWER DISSIPATED PER NODE (P) AND COMPUTATIONAL EFFICIENCY (MMAC/$\mu$W) FOR DIFFERENT IMPLEMENTATIONS DESIGNATED BY DIFFERENT CAPACITIVE LOADS (C). $F_{\text{corner}} = 1$kHz, RESULTING IN SAMPLE RATE OF 10kHz FOR A DIGITAL SYSTEM.

Implementation across process and platforms primarily depends upon the load capacitance resulting from a given implementation. The resulting power dissipated in an FPAA structure is still significantly less than a digital implementation, where a custom configurable fabric or custom implementation, with lower load capacitance, yields significantly lower power dissipated (Table 1). For the typical load capacitance in a configurable structure in 0.13$\mu$m process, we see a power efficiency improvement greater than 1 million over power-efficient digital processors, significantly more efficiency than Mead’s hypothesis of 1000 to 10000 of custom analog IC implementations versus digital implementations [11]. This approach effectively use of the silicon medium by using a minimal number of transistors for this computation and minimizing the required amount of charge moved, and resulting voltage changes. The resulting input currents (Table 1) can be handled and programmed using floating-gate devices independent of transistor threshold voltages. The resulting power dissipation must also consider the required power for the WTA computation; the branches are responsible for most of the computations in the system.

V. CONNECTIONS WITH DENDRITIC COMPUTATION

In this section, we will discuss the connection between a dendrite element and a Hidden Markov Model (HMM) classifier branch, as well as a network of dendrites and somas to create an HMM classifier typical of what is used in speech recognition systems. Figure 4 shows some of the similarities between biological channel populations (Fig. 4a), which we refer as channels throughout, and MOSFET channel populations (Fig. 4b). In both cases, we have a gate modulating the channel between two electrical points, that is inside to outside for the biological channels and source to drain for the MOSFET channel (Fig. 4c) [12], [13]. The transistor circuit in Fig. 1 and Fig. 3 can be used to model dendritic cables, and can be extended into two-dimensional programmable interconnect structures capable of modeling branching dendrites [14]. One can directly build more custom versions of these dendritic elements building CABs geared towards building neuro-inspired and neuro-mimetic systems and still enable reconfigurable capability [14].

Using concepts from IC transistor modeling of dendrites and the IC implementation of an HMM network, we can
reformulate a given HMM tree, as seen in Fig. 5. A diffusive network can be biased such that it is a low-spreading wave-propagation network, typical of dendrites with an increasing diameter, is achieved electronically using programmable diffusor elements. The inputs \( b_i(t) \) to the HMM network set leakage conductances to decrease \( \phi(t) \) over time, which is implemented by allowing \( b_i(t) \) to be applied as an input to part of the programmable diffusor network, and is similar to a transistor channel model of an inhibitory synapse using a Cl reversal potential. Many of the external inputs to a layer of cortical pyramidal cells are inhibitory. The starting inputs at the beginning of the HMM branch are similar to the large number of excitatory synapses at the distal end of the dendritic tree.

Further, one can build a connection between an HMM network and a larger structure of a group of cortical cells. Each dendrite network has a region of computing the starting signal in a typically large network, as well as a region bringing inputs into the dendrite to modulate the behavior. The outputs of these dendrites go into the soma of the neuron, and if large enough, initiate an action potential. These action potentials aggregate on interneurons, which correspondingly decrease the strength of the dendrite outputs, similar to the WTA network. These biological blocks show a striking similarity to how the HMM classifiers are built in this paper. Adding positive feedback through circuits that model Na channels helps renormalize the signal level. Further, we see significant comparisions to HMM classifier networks, which will be elaborated in further discussions. In particular, the selection of a particular HMM branch, and therefore its corresponding symbol location, is similar to the Winner-Take-All (WTA) networks between the excitatory pyramidal cells and other inhibitory cells in cortex, and would be implemented by a range of WTA circuits. The resetting function also has strong correlations to the resulting feedback between cortical neurons in cortex. The comparison between existing HMM algorithms and models of dendritic computation opens many opportunities in both areas.

Fig. 4. Connection between biological channels and MOSFET silicon channels. (a) Cross-Section of a Biological channel, with ions moving through the channel. (b) Cross-Section of a MOSFET with electrons moving through the channel. (c) Band Diagram looking through the channel of the MOSFET. A similar Band Diagram is seen looking through a Biological Channel.

Fig. 5. Comparison of HMM branch with dendritic computation for a neuron. Branch starting circuits are excitatory inputs, that in general are capable of complex input aggregation, both in HMM networks and neurons. For our initial examples, the starting signal is simplified to a single excitatory synapse, but can be directly extended to these approaches.

REFERENCES